РЕФЕРАТ

Отчет 16 страниц, 7 рисунков, 1 таблица.

ГЛАВНЫЙ УЗЕЛ, ВТОРИЧНЫЙ УЗЕЛ, СИНХРОННЫЙ МНОЖЕСТВЕННЫЙ ДОСТУП, ОКНО, ПАКЕТ ДАННЫХ, СИСТЕМА С МНОЖЕСТВЕННЫМ ДОСТУПОМ

Объектом исследования является вычислительная система с множественным доступом.

Цель исследования – изучение процедур и логики реализации процесса управления обмена данными в вычислительной системе с множественным доступом.

В процессе работы была разработана структурная схема вычислительной системы с множественным доступом, также был разработана схема алгоритма программы – имитатора, обеспечивающего процесс работы такой системы. Была проанализирована область применения такой системы.

Областью применения системы с множественным доступом могут быть регистрация в больнице, банкоматы в банковских учреждениях.

СОДЕРЖАНИЕ

|  |  |
| --- | --- |
| Введение………………………………………………………………………. | 4 |
| Основная часть………………………………………………………………... | 5 |
| 1 Структурная схема вычислительной системы с синхронным………... множественным доступом | 6 |
| 2 Схема алгоритма программы – имитатора…………………...………... | 8 |
| Заключение…………………………………………………………………..... | 15 |
| Список использованных источников………………………………………... | 16 |

ВВЕДЕНИЕ

Вычислительные системы с множественным доступом являются разновидностью вычислительных систем с временным уплотнением канала связи. Множественный доступ выражается в требовании одновременной обработки нескольких заданий. Система с множественным доступом состоит из главного узла (эталонного узла) и множества вторичных узлов. Функция главного узла – прием запросов от вторичных узлов на передачу данных. В идеальном случае главный узел должен рассматривать все ожидающие пакеты, как стоящие в одной очереди, которую необходимо обслужить, используя соответствующую дисциплину обслуживания. Но в главном узле нет информации о том, какие вторичные узлы имеют пакеты. И вторичные узлы не имеют информации, а том, какие другие вторичные узлы имеют пакеты. Особенность данной проблемы состоит в том, что информация о состоянии очередей является распределенной.

Передача данных может осуществляться в синхронном и несинхронном режиме.

Системы с несинхронным множественным доступом отличаются от синхронных тем, что в них нет временного окна. Здесь узел, получив пакет данных, сразу передает его, а не ждет начала временного окна. Если пакет попадает в конфликт, то он передается со случайной предварительной задержкой. Такую систему целесообразно использовать, когда пакеты имеют разную длину и малую скорость поступления.

ОСНОВНАЯ ЧАСТЬ

В данной лабораторной работе рассматривается система с синхронным множественным доступом.

В системах с синхронным множественным доступом все передаваемые пакеты данных имеют одинаковую длину, и каждый пакет данных требует для передачи данных единичный интервал времени, который называется окном. Все передающие узлы синхронизированы так, что приём любого пакета данных начинается и заканчивается в пределах окна. Если два или более узла одновременно передают пакеты данных, то возникает конфликт, и главный узел не получает информации. Пакет данных принимается только в том случае, если только один узел передает пакет данных во временном окне. В конце каждого временного окна каждый узел получает ответ от главного узла о потере или получении пакета данных. Каждый пакет, попавший в конфликт должен передаваться повторно. При повторной передаче пакета данных узел пропускает некоторое случайное число окон. Передачи повторяются до тех пор, пока пакет не будет передан в главный узел и не будет им принят. Узел, который отправляет повторно пакет данных, имеет задолженность.

Существует множество способов, используя которые, можно делать выбор между тем, передавать или не передавать в последующих окнах пакеты от узлов, попавших в конфликт. Общей особенностью этих способов является то, что множество вступивших конфликт узлов разбивается на подмножество, причем один узел совершает повторную отправку пакета в следующем окне, а остальные находятся в состоянии ожидания передачи. Если конфликт не разрешился, например, в том случае, если каждый вступивший в конфликт узел находится в одном и том же множестве, то есть несколько узлов отправляют повторно пакет данных в следующем окне, то происходит дальнейшее разбиение на подмножества. Такие алгоритмы называются алгоритмами разбиения.

**1 Структурная схема вычислительной системы с синхронным множественным доступом**

Вычислительные система с множественным доступом состоит из центральной ЭВМ и *N* терминалов. Пакеты данных *S[N]* поступают на передачу в *N-*ыйтерминал через интервалы () единиц условного времени. Для хранения поступающих пакетов данных каждый терминал имеет буфер *Q[N]* (входную память) неограниченной емкости. Из буфера *Q[N]* пакеты данных поступают в ЭВМ на обработку. *G(F)* – функция занятости ЭВМ, которая показывает занята или свободна ЭВМ, а *F* – флаг занятости, показывающий номер терминала, обработкой пакета с которого занята ЭВМ. Функция занятости ЭВМ определяется следующим соотношением:

Если пакеты имеют одно время поступления, то они вступают в конфликт и передаются повторно в *k-*ом последующем окне, где *k* выбирается с равной вероятностью из множества целых чисел, удовлетворяющих условию 1 < *k* <K.Длина окна равна Δt единицам условного времени. Для данного режима работы системы необходимо найти среднюю задержку передачи пакетов как функцию от *k.* Правильность функционирования системы отражает формула (12), которая называется балансом системы.

(1)

В формуле (12) A[i] – количество обработанных заявок.

Это описание соответствует работе структурной схемы вычислительной системы с множественным доступом для синхронного режима передачи пакетов данных (Рисунок 1.1).

 Рисунок 1.1 Структурная схема вычислительной системы с синхронным множественным доступом

**2 Схема алгоритма программы – имитатора**

Тип, диапазон и описание переменных, использованных в схеме алгоритма программы – имитатора, обеспечивающего моделирование процесса функционирования системы представлены в таблице 2.1.

|  |  |  |  |
| --- | --- | --- | --- |
| **Переменная** | **Тип** | **Диапазон** | **Описание** |
| T | float | [0; 3,402E38] | Модельное время |
| Δt | float | [0; 3,402E38] | Приращение модельного времени |
| T | float | [0; 3,402E38] | Время проведения исследования |
| ΔТ | float | [0; 3,402E38] | Период отображения информации о состоянии системы |
| N | ushort | [0; 65535] | Количество терминалов |
| S[N] | ushort | [0; 65535] | Количество заявок, пришедших на терминал |
| Q[N] | ushort | [0; 65535] | Количество заявок в очереди терминала |
| R[N] | ushort | [0; 65535] | Количество обработанных заявок каждого терминала |
| F | ushort | [0; N] | Флаг занятости ЭВМ |
| P[N] | byte | [0; 1] | Наличие задержки у терминала |
| Tп[N] | float | [0; 3,402E38] | Среднее время поступления заявок на терминал |
| Δtп[N] | float | [0; 3,402E38] | Вероятное отклонение поступления заявки |
| tпост[N] | float | [0; 3,402E38] | Время поступления заявки |
| ΔTобр[N] | float | [0; 3,402E38] | Среднее время обработки заявки с терминала |
| Δtобр[N] | float | [0; 3,402E38] | Вероятное отклонение обработки заявки с терминала |
| Tобр[N] | float | [0; 3,402E38] | Время обработки заявки |
| K | byte | [0; 255] | Крайняя граница диапазона для выбора окна задержки |
| Tзад [N] | float | [1; Δt\*K] | Время задержки |
| massindex[N] | ushort | [0; N] | Массив индексов терминалов, попавших в конфликт |
| RAND | float | [0; 1] | Случайное число с плавающей точкой |
| RANDINT | ushort | [0; K] | Целое случайное число |

Таблица 2.1 – Спецификация переменных алгоритма

Схема алгоритма программы – имитатора, обеспечивающего моделирование процесса функционирования системы с множественным доступом (Рисунок 2.1 – Рисунок 2.6).



Рисунок 2.1 – Первый лист схемы алгоритма программы – имитатора



Рисунок 2.2 – Второй лист схемы алгоритма программы – имитатора



Рисунок 2.3 – Третий лист схемы алгоритма программы – имитатора



Рисунок 2.4 – Четвертый лист схемы алгоритма программы – имитатора



Рисунок 2.5 – Пятый лист схемы алгоритма программы – имитатора



Рисунок 2.6 – Шестой лист схемы алгоритма программы – имитатора

ЗАКЛЮЧЕНИЕ

В ходе выполнения лабораторной работы была создана структурная схема вычислительной системы с множественным доступом для синхронной передачи пакетов данных, а также была разработана схема алгоритма программы – имитатора, обеспечивающего процесс функционирования такой системы.

СПИСОК ИПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 7.32 – 2001 СИБИД. Отчет о научно – исследовательской работе. Структура и правила оформления. Взамен ГОСТ 7.32 – 91; Введен с 01.07.2002. Минск, 2002. – 27 с.

2. ГОСТ 7.82 – 2001. Библиографическая запись. Библиографическое описание электронных ресурсов. Минск: Издательство стандартов, 2001. – 26 с.

3. Лоу А.М., Кельтон В.Д. имитационное моделирование. – 3-е изд. – Спб.: Питер, 2004. – 848 с.

4. Якимов В.Н. Имитационное моделирование системс дискретными событиями: лабораторный практикум. – Самара: Самарский государственный технический университет, 2013. – 66 с.